

AMENDMENT

Please amend the above-identified application as follows:

Amendments to the Claims:

This listing of claims will replace all prior versions, and listing, of claims in the application.

1. (Previously Presented): A method for building a representation of a graphical user interface (GUI), comprising:

generating a class, the class being an object oriented programming language class;
generating a first representation of the GUI, wherein the class can produce a second representation GUI based on the first representation;
generating a second representation of the GUI from the class, wherein the second representation includes at least one control; and
wherein the first representation includes hierarchical relationships among controls, control properties, and control event information;

wherein the hierarchal relationships between controls are defined by a control tree that includes a portal control, a desktop control, a page control and a portlet control.

2. (Original): The method of claim 1, further comprising: creating the first representation by parsing a file.

3. (Original): The method of claim 2 wherein:
the file is a JavaServer Pages (JSP) file.

4. (Original): The method of claim I wherein: the second representation is a tree.

5. (Original): The method of claim 1 wherein:

the step of generating the class occurs as a result of receiving a request.

6. (Original): The method of claim 5 wherein:
the request is a hypertext transfer protocol request (HTTP); and the request originates from a web browser.

7. (Original): The method of claim 1, further comprising: providing a response to a web browser.

8. (Original): The method of claim 1 wherein:
the second representation can be driven through at least one lifecycle stage by an interchangeable lifecycle component.

9. (Original): The method of claim 1 wherein:
the at least one control has an interchangeable persistence mechanism.

10. (Original): The method of claim I wherein:
the at least one control can render itself according to a theme.

11. (Original): The method of claim I wherein:
one of the at least one controls can interact with another of the at least one controls.

12. (Original): The method of claim 1 wherein:
one of the at least one controls can advance through the at least one lifecycle stage in parallel with another of the at least one controls.

13. (Original): The method of claim 8 wherein:

the at least one lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose; and wherein the lifecycle stage is part of a dynamically configurable lifecycle.

14. (Original): The method of claim 7 wherein:
the response is a hypertext transfer protocol (HTTP) response.

15. (Original): The method of claim 1 wherein:
the at least one control can raise events and respond to events.

16. (Original): The method of claim 1 wherein:
the at least one control can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.

17. (Previously Presented): A method for building a representation of a graphical user interface (GUI), comprising:
generating a representation of the GUI from metadata, wherein the representation includes at least one control;
driving the representation through at least one lifecycle stage by an interchangeable lifecycle component;
wherein the metadata can include at least one of: hierarchical relationships among controls, control properties, and control event information; and
wherein the representation can be driven through the at least one lifecycle stage by an interchangeable lifecycle component;
wherein an interface is provided to isolate details of the interchangeable lifecycle component from a control container to allow different lifecycle implementations to be

interchangeable, the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls;

wherein the interchangeable life cycle component runs on at least one processor.

18. (Original): The method of claim 17, further comprising:
creating the metadata by parsing a file.

19. (Original): The method of claim 17 wherein:
the step of generating the metadata representation occurs as a result of receiving a request.

20. (Original): The method of claim 19 wherein:
the request is a hypertext transfer protocol request (HTTP); and wherein the request originates from a web browser.

21. (Original): The method of claim 17, further comprising:
providing a response to a web browser.

22. (Original): The method of claim 17 wherein:
the at least one control has an interchangeable persistence mechanism.

23. (Original): The method of claim 17 wherein:
the at least one control can render itself according to a theme.

24. (Original): The method of claim 17 wherein:
one of the at least one controls can interact with another of the at least one controls.

25. (Original): The method of claim 17 wherein:

one of the at least one controls can advance through the at least one lifecycle stage in parallel with another of the at least one controls.

26. (Original): The method of claim 17 wherein:

the at least one lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose; and wherein the lifecycle stage is part of a dynamically configurable lifecycle.

27. (Original): The method of claim 21 wherein:

the response is a hypertext transfer protocol (HTTP) response.

28. (Original): The method of claim 17 wherein:

controls can raise events and respond to events.

29. (Original): The method of claim 17 wherein:

the at least one control can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.

30. (Previously Presented): A system for building a representation of a graphical user interface (GUI), comprising:

a first component operable to produce a second component and a metadata representation of the GUI;

the second component operable to produce a hierarchical representation of the GUI based on the metadata, wherein the representation includes at least one control;

wherein the metadata can include at least one of: hierarchical relationships among controls, control properties, and control event information; and

wherein the representation can be driven through at least one lifecycle stage by an interchangeable lifecycle component;

wherein an interface is provided to isolate details of the interchangeable lifecycle component from a control container to allow different lifecycle implementations to be interchangeable;

wherein the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls; and

wherein the interchangeable life cycle component runs on at least one processor.

31. (Original): The system of claim 30, further comprising:
a parser operable to parse a file and create the metadata.

32. (Original): The system of claim 31 wherein:
the file is a JavaServer Pages (JSP) file.

33. (Previously Presented): The system of claim 30 wherein:
a first generator produces a second generator in response to receiving a request.

34. (Original): The system of claim 33 wherein:
the request is a hypertext transfer protocol request (HTTP); and
wherein the request originates from a web browser.

35. (Original): The system of claim 30 wherein:
a response is provided to a web browser.

36. (Original): The system of claim 30 wherein:
the at least one control has an interchangeable persistence mechanism.
37. (Original): The system of claim 30 wherein:
the at least one control can render itself according to a theme.
38. (Original): The system of claim 30 wherein:
one of the at least one controls can interact with another of the at least one controls.
39. (Original): The system of claim 30 wherein:
one of the at least one controls can advance through the at least one lifecycle stage in parallel with another of the at least one controls.
40. (Original): The system of claim 30 wherein:
the at least one lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose; and
wherein the lifecycle stage is part of a dynamically configurable lifecycle.
41. (Original): The system of claim 35 wherein:
the response is a hypertext transfer protocol (HTTP) response.
42. (Original): The system of claim 30 wherein:
the at least one control can raise events and respond to events.
43. (Original): The system of claim 30 wherein:

the at least one control can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.

44. (Previously Presented): A system comprising:

- a means for generating a first representation of a graphical user interface (GUI);
- a means for generating a second representation of the GUI from the first representation, wherein the second representation includes at least one control;

- wherein metadata can include at least one of hierarchical relationships among controls, control properties, and control event information; and

- wherein the second representation can be driven through at least one lifecycle stage by an interchangeable lifecycle component;

- wherein an interface is provided to isolate details of the interchangeable lifecycle component from a control container to allow different lifecycle implementations to be interchangeable;

- wherein the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls; and

- wherein the interchangeable life cycle component runs on at least one processor.

45. (Original): The system of claim 44, further comprising:

- a means for parsing a file and creating the metadata.

46. (Original): The system of claim 45 wherein:

- the file is a JavaServer Pages (JSP) file.

47. (Original): The system of claim 44, further comprising:

the means for accepting a request.

48. (Original): The system of claim 47 wherein:
the request is a hypertext transfer protocol request (HTTP); and
wherein the request originates from a web browser.
49. (Original): The system of claim 44, further comprising:
a means to provide a response to a web browser.
50. (Original): The system of claim 44 wherein:
the at least one control has an interchangeable persistence mechanism.
51. (Original): The system of claim 44 wherein:
the at least one control can render itself according to a theme.
52. (Original): The system of claim 44 wherein:
one of the at least one controls can interact with another of the at least one controls.
53. (Original): The system of claim 44 wherein:
one of the at least one controls can advance through the at least one lifecycle stage in
parallel with another of the at least one controls.
54. (Original): The system of claim 44 wherein:
the at least one lifecycle stage is one of: init, load state, create child controls, load, raise
events, pre-render, render, save state, unload and dispose; and wherein the lifecycle stage is part
of a dynamically configurable lifecycle.

55. (Original): The system of claim 49 wherein:
the response is a hypertext transfer protocol (HTTP) response.
56. (Original): The system of claim 44 wherein:
the at least one control can raise events and respond to events.
57. (Original): The system of claim 44 wherein:
the at least one control can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.
58. (Previously Presented): A machine readable medium having instructions stored thereon that when executed by a processor cause a system to:
generate a representation of a graphical user interface (GUI) from metadata, wherein the representation includes at least one control;
drive the representation through at least one lifecycle stage by an interchangeable lifecycle component;
wherein the metadata can include at least one of: hierarchical relationships among controls, control properties, and control event information; and
wherein the representation can be driven through the at least one lifecycle stage by the interchangeable lifecycle component;
wherein an interface is provided to isolate details of the interchangeable lifecycle component from a control container to allow different lifecycle implementations to be interchangeable;
wherein the interchangeable lifecycle component further drives a control tree through different lifecycle stages, the control tree includes portal controls, desktop controls, page controls and portlet controls; and

wherein the interchangeable life cycle component runs on at least one processor.

59. (Original): The machine readable medium of claim 58, further comprising instructions that when executed cause the system to:

create the metadata by parsing a file.

60. (Original): The machine readable medium of claim 58 wherein:

the step of generating the metadata representation occurs as a result of receiving a request.

61. (Original): The machine readable medium of claim 60 wherein:

the request is a hypertext transfer protocol request (HTTP); and wherein the request originates from a web browser.

62. (Original): The machine readable medium of claim 58, further comprising instructions that when executed cause the system to:

provide a response to a web browser.

63. (Original): The machine readable medium of claim 58 wherein:

the at least one control has an interchangeable persistence mechanism.

64. (Original): The machine readable medium of claim 58 wherein:

the at least one control can render itself according to a theme.

65. (Original): The machine readable medium of claim 58 wherein:

one of the at least one controls can interact with another of the at least one controls.

66. (Original): The machine readable medium of claim 58 wherein:
one of the at least one controls can advance through the at least one lifecycle stage in parallel with another of the at least one controls.
67. (Original): The machine readable medium of claim 58 wherein:
the at least one lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose; and wherein the lifecycle stage is part of a dynamically configurable lifecycle.
68. (Original): The machine readable medium of claim 62 wherein:
the response is a hypertext transfer protocol (HTTP) response.
69. (Original): The machine readable medium of claim 58 wherein:
controls can raise events and respond to events.
70. (Original): The machine readable medium of claim 58 wherein:
the at least one control can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.
71. (Canceled)